

# MANUAL PRÁCTICO DE HTML

© 1995 Álvaro Martínez Echevarría  
Escuela Técnica Superior de Ingenieros de Telecomunicación  
Universidad Politécnica de Madrid, España



---

## NOTA PREVIA

La primera versión de este trabajo se desarrolló en el marco de mi trabajo como becario en la Escuela de Ingenieros de Telecomunicación, durante mayo de 1995. Tanto su creación como su posterior mantenimiento son una aportación personal desinteresada a la comunidad hispanohablante de Internet, y no están ligadas a ningún interés comercial.

No está permitido copiar o modificar este manual sin autorización expresa de su autor (o sea, yo :-). Se permite hacer enlaces a este manual desde páginas personales y comerciales, con la única restricción de que en las páginas de origen de dichos enlaces no haya lugar a confusión sobre la autoría y motivación de este manual.

---

## INTRODUCCIÓN

Este manual pretende ser una introducción básica al lenguaje HTML, que permite escribir páginas de WWW. Su orientación es más bien práctica, por lo que no se han tenido en cuenta cuestiones como las diferentes versiones de HTML a las que pertenece cada directiva (lo cual resulta a veces más importante de lo que parece, porque implica que algunas directivas sólo funcionen con los programas de WWW más modernos: por ejemplo `<center>`).

Según creo, esta es una de las pocas (o la única) referencias sobre HTML escritas en castellano (que alguien me corrija si me equivoco). Para referencias más amplias (aunque en inglés) ver el final de esta página.

Mi agradecimiento a todas las personas que me han hecho llegar sus opiniones y consejos acerca de este manual. En sucesivas versiones iré mejorándolo para sea todo lo útil y completo que espero que sea. Si quieres dejarme algún mensaje, puedes mandar correo electrónico a [alvaro@etsit.upm.es](mailto:alvaro@etsit.upm.es).

---

## CONTENIDOS

- Qué es HTML
- Cómo especificar efectos del texto
- Estructura básica de un documento HTML
- Estilos y efectos básicos
  - Títulos

- Atributos del texto
  - Listas
  - Varios
  - Enlaces y gráficos
    - Introducción
    - Qué es un URL
    - Enlaces
    - Gráficos
  - Caracteres especiales
  - Referencias
- 

## QUÉ ES HTML

HTML (*HyperText Markup Language*) es un lenguaje muy sencillo que permite describir hipertexto, es decir, texto presentado de forma estructurada y agradable, con *enlaces* (*hyperlinks*) que conducen a otros documentos o fuentes de información relacionadas, y con *inserciones* multimedia (gráficos, sonido...) La descripción se basa en especificar en el texto la estructura lógica del contenido (títulos, párrafos de texto normal, enumeraciones, definiciones, citas, etc) así como los diferentes efectos que se quieren dar (especificar los lugares del documento donde se debe poner cursiva, negrita, o un gráfico determinado) y dejar que luego la presentación final de dicho hipertexto se realice por un programa especializado (como Mosaic, o Netscape).

---

## CÓMO ESPECIFICAR EFECTOS DEL TEXTO

La mayoría de los efectos se especifican de la misma forma: rodeando el texto que se quiere marcar entre dos *etiquetas* o *directivas* (*tags*, en inglés), que definen el efecto o unidad lógica que se desea. Las etiquetas están formadas por determinados códigos metidos entre los signos < y >, y con la barra / cuando se trata de la segunda etiqueta de un efecto (la de cierre). Por ejemplo: <efecto> para abrir y </efecto> para cerrar. Ciertas directivas sólo se ponen una vez en el lugar del texto donde queramos que aparezca el efecto concreto. Esto es lo que ocurre, por ejemplo, cuando queremos poner un gráfico, caso en el que se usa algo parecido a <poner\_gráfico\_aquí> (más adelante ya veremos la directiva concreta que se utiliza).

A veces es necesario ofrecer datos adicionales en una directiva. Por ejemplo, cuando se define un *hiperenlace* hay que especificar su destino. Para ello se incluyen parámetros en la directiva inicial (la de apertura), de la siguiente forma: <efecto parametro1 parametro2 . . .>. La directiva de cierre, caso de ser necesaria, queda como antes: </efecto>.

Más adelante en el presente documento se muestra el efecto de las directivas más usadas en la creación de un documento HTML. Para cada una de ellas, primero se muestra el texto fuente, y bajo éste, el efecto que produce.

---

## ESTRUCTURA BÁSICA DE UN DOCUMENTO HTML

Un documento HTML comienza con la etiqueta `<html>`, y termina con `</html>`. Dentro del documento (entre las etiquetas de principio y fin de html), hay dos zonas bien diferenciadas: el *encabezamiento*, delimitado por `<head>` y `</head>`, que sirve para definir diversos valores válidos en todo el documento; y el *cuerpo*, delimitado por `<body>` y `</body>`, donde reside la información del documento.

La única utilidad del encabezamiento en la que nos detendremos es la directiva `<title>`, que permite especificar el título de un documento HTML. Este título no forma parte del documento en sí: no aparece, por ejemplo, al principio del documento una vez que este se presenta con un programa adecuado, sino que suele servir como título de la ventana del programa que nos la muestra. Por ejemplo, en el encabezamiento de este manual se ha especificado:

```
<title>Manual práctico de HTML</title>
```

en minúsculas. Obsérvese que el título que encabeza este texto se ha escrito con mayúsculas, para distinguirlo del título *global* del documento.

El cuerpo de un documento HTML contiene el texto que, con la presentación y los efectos que se decidan, se presentará ante el *hiperlector*. Dentro del cuerpo son aplicables todos los efectos que se van a mencionar en el resto de esta guía. Dichos efectos se especifican exclusivamente a través de directivas. Esto quiere decir que los espacios, tabulaciones y retornos de carro que se introduzcan en el fichero fuente no tienen ningún efecto a la hora de la presentación final del documento. Por ejemplo, escribiendo:

```
Estas
      palabras
forman          una
      frase.
```

producimos exactamente lo mismo que con:

```
Estas palabras forman una frase.
```

A la hora de la verdad lo que se ve es:

```
Estas palabras forman una frase.
```

En resumen, la estructura básica de un documento HTML queda de la forma siguiente:

```
<html>
<head>
<title>Título</title>
</head>
<body>
Texto del documento, menciones a gráficos, enlaces, etc.
</body>
</html>
```

---

## ESTILOS Y EFECTOS BÁSICOS

Como ya hemos dicho, la estructura lógica del texto y los diferentes efectos que se le apliquen se especifican mediante *directivas*. En este punto vamos a repasar algunas de las más importantes. En cada uno de los casos que veremos, primero se presenta el texto original HTML, es decir, lo que nosotros editamos, con las directivas situadas en los lugares adecuados; y después se presenta el efecto que dicho *texto fuente* produce una vez que se interpreta y se representa con el programa adecuado.

---

### TÍTULOS

Mediante los títulos, en sus diferentes niveles de importancia, podemos definir el *esqueleto* del documento, su estructura básica.

```
<h1>Mucha importancia</h1>
```

## Mucha importancia

```
<h2>Menos importancia</h2>
```

### Menos importancia

```
<h3>Mucha menos importancia</h3>
```

#### Mucha menos importancia

---

### ATRIBUTOS DEL TEXTO

Mediante estos atributos determinamos el estilo y el tipo de letra que tendrá la presentación del documento final.

El primero en el que nos deberíamos detener es el *texto normal* entendiendo como tal el que no tiene ninguna característica especial. Para definir un párrafo como *normal* no es necesario poner ninguna etiqueta. Lo único que hay que tener en cuenta, como ya se ha dicho antes, es que al presentar el documento se hace caso omiso de los espacios, tabulaciones y retornos de carro que se encuentren en el texto fuente. Por ello cuando se quiera forzar un final de línea es necesario utilizar dos directivas especiales: `<p>` para marcar un fin de párrafo, y `<br>` para un único retorno de carro. La diferencia entre ambas es que la separación de líneas que provoca `<p>` es algo mayor que la de `<br>`, para que los párrafos se distingan bien entre sí. Las dos directivas mencionadas se sitúan en el punto en que queremos poner la separación. Por ejemplo:

```
Este será un texto normal (párrafo 1, línea 1).<br>
El primer párrafo estará formado por 2 líneas (párrafo 1, línea 2).<p>
Este ya es el segundo párrafo (párrafo 2, línea 1).<p>
```

Este será un texto normal (párrafo 1, línea 1).

El primer párrafo estará formado por 2 líneas (párrafo 1, línea 2).

Este ya es el segundo párrafo (párrafo 2, línea 1).

Por supuesto, estas dos etiquetas se puede aplicar donde queramos, no sólo en el texto normal.

El *texto preformateado* (etiqueta `<pre>`) se aplica cuando queremos que en la presentación final del documento se respeten los espacios y retornos de carro que hayamos puesto en el texto fuente. Además se utilizará un tipo de letra de espaciado fijo, parecido al de una máquina de escribir, más pequeño que el del texto normal. Este estilo de texto puede ser adecuado, por ejemplo, para una tabla numérica sencilla:

```
<pre>
Texto preformateado
-----
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
-----
</pre>
```

```
Texto preformateado
-----
| 1 | 2 | 3 | 4 |
| 5 | 6 | 7 | 8 |
| 9 | 10 | 11 | 12 |
-----
```

Para hacer una cita textual dentro de nuestro documento, se puede utilizar la directiva `<blockquote>`:

```
<blockquote>Muchos años después, frente al pelotón de fusilamiento,
el coronel Aureliano Buendía había de recordar aquella tarde remota
en que su padre lo llevó a conocer el hielo.<br>
(Gabriel García Márquez, Cien años de soledad)</blockquote>
```

Muchos años después, frente al pelotón de fusilamiento, el coronel Aureliano Buendía había de recordar aquella tarde remota en que su padre lo llevó a conocer el hielo.  
(Gabriel García Márquez, Cien años de soledad)

Las *direcciones* de correo electrónico se suelen marcar con esta directiva:

```
<address>Dirección: webmaster@etsit.upm.es</address>
```

*Dirección: webmaster@etsit.upm.es*

Se pueden dar también los atributos más tradicionales: negrita y cursiva:

```
<b>Esto en negrita</b> y <i>esto en cursiva</i>
```

**Esto en negrita** y *esto en cursiva*

Se puede utilizar un tipo de letra similar al de una máquina de escribir:

```
<tt>Máquina de escribir</tt>
```

Máquina de escribir

Para centrar texto (o, en general, cualquier cosa: un gráfico, por ejemplo) se usa la directiva `<center>`:

```
<center>Verde que te quiero verde</center>
```

Verde que te quiero verde

---

## LISTAS

Las listas se definen de forma muy sencilla: se dice dónde empieza la lista, dónde empieza cada punto y dónde acaba la lista. Las etiquetas que se utilicen en cada caso deben aparecer al principio de línea, o al menos sin texto por delante (sólo espacios o tabulaciones).

Podemos recurrir a tres tipos distintos de listas, cada una con una presentación diferente: no numeradas, numeradas y listas de definiciones (glosarios).

Las listas se pueden anidar, es decir, en el lugar donde debería ir uno de los términos de la lista se pone una nueva lista, que por supuesto no tiene porqué ser del mismo tipo.

Esto es una lista no numerada:

```
<ul>
<li>Tomates
<li>Zanahorias
<li>Puerros
</ul>
```

- Tomates
- Zanahorias
- Puerros

Esto una lista numerada:

```
<ol>
<li>Miguel Induráin
<li>Tony Rominger
<li>Eugeni Berzin
</ol>
```

1. Miguel Induráin
2. Tony Rominger
3. Eugeni Berzin

Un glosario está formado por una serie de parejas de *término* (marcado con `<dt>` al principio de línea) y *definición* (con `<dd>`). Por ejemplo, podríamos crear un pequeño diccionario con los términos perro, gato y pescadilla, de la siguiente manera:

```
<dl>
<dt>Perro (<i>n. masc.</i>)
<dd>Animal de cuatro patas que ladra.
<dt>Gato (<i>n. masc.</i>)
<dd>Animal de cuatro patas que maúlla y se lleva muy mal con el perro.
<dt>Pescadilla (<i>n. fem.</i>)
<dd>Animal que vive en el mar y está recubierto de escamas.
</dl>
```

Perro (*n. masc.*)

Animal de cuatro patas que ladra.

Gato (*n. masc.*)

Animal de cuatro patas que maúlla y se lleva muy mal con el perro.

Pescadilla (*n. fem.*)

Animal que vive en el mar y está recubierto de escamas.

---

## VARIOS

La directiva `<hr>` sitúa en el documento una línea horizontal de separación. En este documento, por ejemplo, se han utilizado líneas horizontales para separar las diferentes secciones:

```
<hr>
```

---

Para poner un comentario en un documento HTML, es decir, una aclaración que no aparece en la presentación final del documento, se encierra el texto que formará el comentario entre los símbolos `<!--` y `-->`. Por ejemplo, un caso típico podría ser:

```
<!-- Modificado por Álvaro el viernes 2 de junio -->
```

---

## ENLACES Y GRÁFICOS

### INTRODUCCIÓN

Además de los muchos estilos y capacidades de presentación que nos ofrece HTML para estructurar el documento en sí, disponemos de varias directivas que nos permiten definir relaciones entre diferentes documentos y estructurar todo un conjunto de documentos para crear una unidad lógica. La facilidad para

definir este tipo de enlaces es una de las razones de la potencia y versatilidad de HTML. Por la similitud de tratamiento que tienen los enlaces y los gráficos, tocaremos también en esta sección cómo pueden incluirse estos últimos en un documento.

Los enlaces en HTML se expresan rodeando con la directiva `<a>` el objeto (que puede ser un fragmento de texto o un gráfico) que vaya a servir como *anclaje* para el enlace. Por ejemplo, si marcamos con `<a>` un gráfico, cuando en el documento final se pulse con el ratón sobre dicho gráfico *saltaremos* al objeto referenciado en el enlace: otro documento, un vídeo musical, o un servidor de información meteorológica.

## QUÉ ES UN URL

Para especificar de manera uniforme el objeto al que apunta nuestro enlace, se utiliza una forma estandarizada que se denomina URL (*Uniform Resource Locator*, es decir, *Localizador Uniforme de Recursos*). Un URL está formado de la siguiente manera: *esquema://maquina/ruta* (en realidad, como se verá dentro de un momento, la barra / puede considerarse parte de la *ruta*).

El *esquema* es un nombre que identifica el tipo de servicio que va a proporcionarse en el destino del enlace. La razón de esta aparente complicación es que el WWW pretende unificar el acceso a servicios de información que previamente eran incompatibles entre sí, como ftp, gopher o telnet. El esquema más utilizado es http, correspondiente al propio WWW (es decir, que cualquier referencia a un documento HTML debería comenzar con `http://`). Otros esquemas muy frecuentes son ftp, telnet, gopher o wais.

La *máquina* y la *ruta* sirven para localizar el objeto al que apunta nuestro enlace. La *máquina* es la identificación del servidor en el cual está situado el objeto al que apunta el enlace. Puede ser simplemente el nombre de un ordenador (como `www.etsit.upm.es`) o también un nombre y un puerto (por ejemplo `www.etsit.upm.es:8000`).

La *ruta* es el nombre del fichero que contiene el documento en concreto, incluyendo el nombre del subdirectorío en el que se encuentra. Los diferentes nombres que constituyan la ruta completa al archivo se deben separar con la barra / (inclinada hacia la derecha), tal y como se hace en el sistema operativo UNIX (y al revés que en MS-DOS). La razón de este convenio es precisamente que la mayor parte de los servidores de WWW que hay en Internet son ordenadores basados en UNIX, debido a la gran superioridad tecnológica de este sistema sobre MS-DOS. Esto se nota también en que por lo general los nombres de los ficheros no tienen muchas limitaciones: pueden ser casi tan largos como queramos, contener varios puntos, etc. Por ejemplo, el nombre de cierto fichero situado en un servidor podría ser `/info/documentos/ciencia/fisica/relatividad.html`. Debemos tener en cuenta que en UNIX las mayúsculas y las minúsculas son distintas en los nombres de los ficheros: no es igual FICHERO que fichero.

Conviene que nos detengamos momentáneamente en la estructuración habitual de los ficheros en un servidor de WWW. Para empezar, siempre hay una *página de bienvenida* (*home page*) que podría compararse con la *portada* de un periódico o revista: si no sabemos exactamente qué es lo que buscamos, o no sabemos dónde encontrarlo, la portada es lo primero que vemos. Para acceder a la *home page* de cualquier servidor de WWW, basta con escribir una barra en el lugar de la ruta (es decir, reclamamos al servidor el *directorio raíz*). Por ejemplo, para acceder a la página de bienvenida de la ETSIT, hay que dirigirse a `http://www.etsit.upm.es/`, y para ir a la de la NASA habría que contactar con `http://www.nasa.gov/`.

El resto de la información que se puede encontrar en un servidor de WWW se distribuye a partir de ese directorio raíz en distintos subdirectoríos y archivos. Un convenio muy habitual relativo al nombre de los ficheros es hacer que los archivos que contengan documentos HTML terminen en `.html`.

## ENLACES

Con lo que ya hemos dicho, podemos abordar sin problemas el asunto que originalmente nos ocupaba: cómo se introducen enlaces en un documento HTML. Para definir un enlace es necesario marcar con la directiva `<a>` el objeto del cual va a partir dicho enlace. Dicha directiva debe incluir el parámetro `href="URL"` para especificar el destino del enlace. Es decir, que antes del objeto elegido debemos abrir con `<a href="URL">`, y después cerrar con `</a>`. Por ejemplo, si queremos que el texto *pulse aquí para visitar la NASA* nos conduzca a la *home page* de la NASA, debemos escribir en nuestro texto HTML:

```
<a href="http://www.nasa.gov/">Pulse aquí para visitar a la NASA</a>
```

Lo cual producirá el resultado:

[Pulse aquí para visitar la NASA](http://www.nasa.gov/)

Por lo general no nos preocupa irnos tan lejos, sino sencillamente enlazar con otro documento que se encuentra en el mismo servidor, puede que incluso que en el mismo subdirectorio. En este caso no es necesario escribir el *camino completo* al destino del enlace, sino que basta con dar la mínima información imprescindible. El programa que se use para leer el documento final suele ser lo bastante *listo* como para deducir el resto. Es decir, que si desde cierto documento queremos enlazar con otro que se encuentra en el mismo subdirectorio, basta con poner su nombre: `<a href="el_otro_fichero">pulse aquí</a>`. O si se encuentra en otro subdirectorio del mismo servidor, es suficiente con poner `<a href="/la/ruta/que/sea/fichero.html">pulse aquí</a>`. También pueden utilizarse rutas relativas: `<a href="ruta/relativa/cosa.html">cosa</a>`.

## GRÁFICOS

Para incluir un gráfico en un documento HTML se utiliza la directiva `<img>`. En dicha directiva debe incluirse un parámetro `src="URL"`, con el cual indicamos dónde está el fichero con el gráfico concreto que queremos para nuestro documento. Esto pone a nuestra disposición una gran flexibilidad, ya que podemos complementar el contenido de nuestro documento tanto con gráficos que se encuentren disponibles en nuestro servidor de WWW como con una foto situada en un servidor de la NASA o del Ministerio de Cultura, por ejemplo, sin que el lector final tenga por qué apreciar ninguna diferencia.

Existe alguna limitación respecto a los formatos gráficos que los programas lectores de HTML puede interpretar sin problemas. El formato fundamental es el GIF, que cualquier programa con capacidades gráficas debería poder mostrar directamente en nuestro texto (Mosaic y Netscape pueden hacerlo). Si utilizamos otro formato diferente, lo más probable es que cuando un lector esté accediendo al documento, el programa no comprenda ese formato y se tenga que solicitar la *ayuda* de otro programa, con lo cual al final el gráfico no se insertará en el lugar estratégico de nuestro documento, sino que aparecerá en otra ventana diferente.

Hay un parámetro optativo de la directiva `<img>` que sirve para proponer un texto alternativo a un gráfico. Este texto aparecerá cuando se esté usando para leer el HTML un programa sin capacidades gráficas (por ejemplo Lynx, que sólo trabaja con texto). Se trata de `alt="texto"`. Conviene utilizarlo cuando los gráficos sirven como origen a *hiperenlaces*, porque si no los programas sin capacidades gráficas no podrían mostrar los enlaces que nosotros queremos.

Como ocurría antes con los enlaces, por lo general no es necesario escribir el URL completo, sino que basta con dar la mínima información. Por ejemplo, para colocar en este punto del documento un monigote que está en el mismo subdirectorio que este manual, en el fichero `monigote.gif`, escribiremos:

```
<p>
```

Lo que se traduce en:



Como se ve, hemos especificado el texto alternativo "MONIGOTE", con lo cual una persona que no dispusiera del programa adecuado hubiera visto algo parecido a [ MONIGOTE ] en lugar del dibujo.

Podemos también incluir un dibujo que esté en otro lugar especificando un URL completo, por ejemplo:

```
<p>
```

Imagen



(<) o el signo inglés de *and* (llamado *ampersand*: &).

Trataremos primero el caso más sencillo. Existe una razón evidente que impide que podamos escribir ciertos símbolos directamente en un texto HTML, como por ejemplo el <: dichos símbolos tienen un significado en HTML, y es necesario diferenciar claramente cuándo poseen ese significado y cuándo queremos que aparezcan literalmente en el documento final. Por ejemplo, como ya sabemos, < indica el comienzo de una directiva, y, por ello, si queremos que aparezca en el texto como tal tendremos que *dar un rodeo* escribiendo algo que no de lugar a confusión, en este caso &lt ; . Los símbolos afectados por esta limitación, y la forma de escribirlos, se detallan a continuación:

- < (*Menor que*): &lt;
- > (*Mayor que*): &gt;
- & (símbolo de *and*, o *ampersand*): &amp;
- " (*comillas dobles*): &quot;

Es decir, que para escribir <"> en nuestro texto HTML original debemos poner &lt ; &quot ; &gt ; .

El otro caso especial se da cuando en un texto HTML se quiere escribir una ñe, por ejemplo. Existen dos formas de hacerlo. La primera, que es a la que obliga el estándar de HTML, consiste en utilizar *entidades*, es decir, *palabrejas* como las que antes se presentaron para escribir ciertos símbolos. Las entidades comienzan siempre con el símbolo &, y terminan con un punto y coma (;). Entre medias va un identificador del carácter que queremos que se escriba. Las entidades necesarias en nuestro idioma son:

- á: &aacute;
- é: &eacute;
- í: &iacute;
- ó: &oacute;
- ú: &uacute;
- Á: &Aacute;
- É: &Eacute;
- Í: &Iacute;
- Ó: &Oacute;
- Ú: &Uacute;
- ü: &uuml;
- Ü: &Uuml;
- ñ: &ntilde;
- Ñ: &Ntilde;
- ¿: &#191;
- ¡: &#161;

Como puede verse, las vocales acentuadas se identifican añadiendo el sufijo *acute* a la vocal sin acentuar (puesto que se trata de un acento agudo). Para la u con diéresis y la ñe se usan *uml* tras una u y *tilde* detrás una ene, respectivamente. La equivalencia de los signos de abrir interrogación y exclamación es algo más oscura: a falta de una denominación más evidente, tenemos que usar el valor numérico de dichos caracteres

en el código estándar latin1 (ISO-8859-1). Esto se puede hacer con cualquier otro carácter del código latin1, que es el código de caracteres básico en HTML, escribiendo `&#numero;`.

La segunda manera, que sin duda es más cómoda, consiste en no preocuparse por esta limitación y escribir literalmente los caracteres afectados. A pesar de que este método suele funcionar en las conexiones WWW directas (porque el protocolo HTTP, que transporta el HTML por los vericuetos de Internet, requiere un canal de 8 bits), no tiene por qué funcionar bien cuando los documentos HTML se envían por correo electrónico, por ejemplo. Por tanto, y a pesar de los inconvenientes, es absolutamente recomendable respetar la norma especificada en HTML.

En cualquier caso, no resulta muy complicado escribir un programa que traduzca todas las apariciones de los caracteres especiales por sus correspondientes entidades HTML, o viceversa. Con un programa así, uno puede escribir los documentos sin preocuparse por estos problemas, y luego traducir a HTML correcto. Al final de este manual encontrará un enlace a un programita en C que hace precisamente eso.

Volviendo al reconocimiento de culpa que antes de hizo por haber ocultado estos *detallitos*, debemos decir que muchos de los ejemplos que antes se pusieron no eran totalmente correctos. Por ejemplo, cuando se escribió:

```
<dd>Animal de cuatro patas que maúlla y se lleva muy mal con el perro.
```

En realidad debería haberse puesto:

```
<dd>Animal de cuatro patas que ma&uacute;lla y se lleva muy mal con el perro.
```

Y en lugar de:

```
<a href="http://www.nasa.gov/">Pulse aquí para visitar a la NASA</a>
```

Debería haberse escrito:

```
<a href="http://www.nasa.gov/">Pulse aqu&iacute; para visitar a la NASA</a>
```

---

## REFERENCIAS

Existen muchas introducciones a HTML; todas las que yo conozco están en inglés. Las más recomendables desde mi punto de vista, son:

- [A Begginer's Guide to HTML](#)
- [The HTML Quick Reference Guide](#)
- [La especificación oficial de HTML](#)

Para una introducción a los *formularios* HTML, que no se han tratado en este manual, se puede revisar [Fill-out Forms Overview](#).

Para más información sobre URLs (*Uniform Resource Locators*), se puede consultar:

- [A Beginner's Guide to URLs](#)
- [WWW Names and Addresses, URIs, URLs, URNs](#)

Quien tenga interés sobre HTTP (*HyperText Transfer Protocol*), puede recurrir a la [especificación oficial de HTTP 1.0](#).

El programa que se mencionó en la sección sobre caracteres especiales se puede encontrar [aquí](#). Permite convertir con comodidad varios caracteres especiales a sus entidades HTML correspondientes.

Podrás encontrar algo más de información en mi página de WWW. Te invito a hacerme una visita, [aquí](#)

---

© Mayo de 1995 [alvaro@etsit.upm.es](mailto:alvaro@etsit.upm.es)